REMARKS/ARGUMENTS

This Amendment is in response to the Final Office Action dated January 20, 2004. Claims 1-10 are pending in the present application. Claims 1-10 have been rejected. Accordingly, claims 1-10 remain pending. For the reasons set forth more fully below, Applicants respectfully submit that the claims as presented are allowable. Consequently, reconsideration, allowance, and passage to issue are respectfully requested.

In the event, however, that the Examiner is not persuaded by Applicants' arguments, Applicants respectfully request that the Examiner enter the arguments to clarify issues upon appeal.

Claim Rejections - 35 U.S.C. §103

The Examiner has stated:

> **Claims 1-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,567,837 to Robinson in view of U.S. Patent No. 6,108,715 to Leach.**
>
> **As to claim 1, Robinson teaches the invention substantially as claimed including remote procedure calls [communications interface 24 allows the array of active objects 23a-23c to communicate with other object oriented processor arrays, Fig.2; col. 7, line 65 – col. 8, line 45] in a multiprocessing system, ...**
>
> **Although the Robinson reference shows substantial features of the claimed invention, it does not explicitly show the reserved address being known to a remote procedure call requestor.**
>
> **However, Leach teaches remote procedure calls [col. 4, lines 31 – 57], reserved address [virtual function table; col. 11, lines 1 – 48], and the reserved address being known [virtual table pointer is used to identify an address of a virtual table 624 that the kernel uses to invoke the real object's methods; col. 7, lines 35 – 54].**
>
> **It would have been obvious to a person of ordinary skilled in the art at the time of the invention to apply the teaching of allowing a remote procedure call requestor to know the location of a reserved address as taught by Leach to the invention of Robinson because the virtual table index in conjunction with the resource table's virtual table pointer is used to discern the actual address of the real method and this determined address allows a jump to the real method's address in order to execute the real method [col. 15, lines 45 – 54 of Leach]...**
>
> **Applicant's arguments with respect to claims 1 – 10 have been considered but are moot in view of the new ground(s) of rejection.**

Applicants respectfully disagree with the Examiner's rejections. For the Examiner's convenience, independent claims 1 and 6 are reproduced in their entirety herein below.

Claims 1 and 6

> 1. (previously amended) A method for providing remote procedure calls in a multiprocessing system, the multiprocessing system including a general purpose processor and a plurality of network processors; each of the plurality of network processors having a memory, the method comprising the steps of:
>
> (a) accessing a reserved address in the memory of at least one of the plurality of network processors, wherein the reserved address comprises a first portion and a second portion, wherein the reserved address is known to a remote procedure call requestor, wherein the second portion comprises a pointer for an instruction address of a procedure code, wherein the instruction address is not known to the remote procedure call requestor;
>
> (b) initiating a software action by the first portion of the reserved address, wherein the software action comprises obtaining the pointer in the second portion of the reserved address; and
>
> (c) accessing and processing the procedure code at the instruction address utilizing the pointer.
>
> 6. (previously amended) A system for providing remote procedure calls in a multiprocessing system, the multiprocessing system including a general purpose processor and a plurality of network processors; each of the plurality of network processors having a memory, the system comprising:
>
> means for accessing a reserved address in the memory of at least one of the plurality of network processors, wherein the reserved address comprises a first portion and a second portion, wherein the reserved address is known to a remote procedure call requestor, wherein the second portion comprises a pointer for an instruction address of a procedure code, wherein the instruction address is not known to the remote procedure call requestor;
>
> means for initiating a software action by the first portion of the reserved address, wherein the software action comprises obtaining the pointer in the second portion of the reserved address; and
>
> means for accessing and processing the procedure code at the instruction address utilizing the pointer.

The present invention provides a method for providing remote procedure calls in a multiprocessing system including a general purpose processor and a plurality of network processors. Each of the plurality of network processors has a memory. The method comprises accessing a reserved address in the memory of at least one of the plurality of network processors, where the reserved address comprises a first portion and a second portion. The reserved address is known to a remote procedure call requestor. The second portion comprises a pointer for an

instruction address of a procedure code, and the instruction address is not known to the remote procedure call requestor. The method further comprises initiating a software action by the first portion of the reserved address, where the software action comprises the step of obtaining the pointer in the second portion of the reserved address. The method further comprises accessing and processing the procedure code at the instruction address utilizing the pointer. (Summary and Figure 6 and on page 6, line 21, to page 7, line 6.) Robinson in view of Leach does not teach or suggest these features, as discussed below.

Robinson discloses an object oriented processor array, which includes a library of functional objects which are instantiated by commands through a system object and which communicate via a high level language. According to one embodiment, the object oriented processor array is formed on a single chip or on a single processor chip and an associated memory chip. When several objects are instantiated on a single chip, pins may be assigned to each object via a high-level command language. (Abstract.)

Leach discloses a method and system that allows a client process to invoke a remote procedure. An operating system maintains a table with an entry for each remote procedure. Each entry of this table contains a signature that specifies a format in which parameters are exchanged between the client process and the remote procedure. When the client process requests the invocation of the remote procedure, the operating system creates a stack for the remote procedure. This stack is then mapped into the operating system's address space. By mapping the remote procedure's stack in this fashion, the operating system can simultaneously access the client's stack and the remote procedure's stack. The operating system then copies, in accordance with the remote procedure's signature, parameters directly from the client's stack to the remote procedure's stack. Once the parameters are copied, the remote procedure executes using the data

contained on its own stack. A kernel, which invokes the remote procedure, determines the address of the remote procedure using a via table index, which is passed in a register to index into a virtual function table. After determining the remote procedure, the kernel jumps to the address to execute the remote procedure. When the substantive execution of the remote procedure is complete, the remote procedure traps back to the operating system. The operating system then copies, in accordance with the remote procedure's signature, return parameters from the remote procedure's stack to the client's stack. Once these parameters have been copied, the operating system returns to the client process so that the client process can continue with its execution. (Abstract, column 4, lines 32-56, column 11, lines 10-13, and lines 45-48, and column 15, lines 44-52.)

Robinson in view of Leach does not teach or suggest the reserved address, "wherein the reserved address comprises a first portion and a second portion, wherein **the reserved address is known to a remote procedure call requestor**, wherein the second portion comprises a pointer for an instruction address of a procedure code, wherein the **instruction address is not known to the remote procedure call requestor**," as recited in independent claims 1 and 6.

Applicants agree with the Examiner that Robinson does not explicitly show a reserved address that is known to a remote procedure call requestor. Applicants respectfully assert that Leach does not teach or suggest the "reserved address," as recited in independent claims 1 and 6. Instead, Leach teaches a "virtual function table," which functions as a jump table. The jump table of Leach is different from the reserved address of the present invention, because the jump table is used by a kernel, which invokes the remote procedure, to "jump to" the address of the remote procedure (column 4, lines 32-56, column 11, lines 10-13, and lines 45-48, and column 15, lines 44-52). The reserved address of the present invention eliminates the need for a jump

table to provide the instruction address for the remote procedure. This is beneficial because a jump table requires pre-allocated memory space and other resources to regularly update the jump table's index (specification, page 1, line 18, to page 2, line 3). Instead of using the reserved address of the present invention, both Robinson and Leach teach **tables** that point to an address of functions or procedures, and these tables require pre-allocated memory space (Robinson, column 9, lines 22-39, and Leach, column 11, lines 10-13). Accordingly, the jump table of Leach cannot be combined with the system of Robinson to provide the reserved address, "wherein the reserved address is known to a remote procedure call requestor," as recited in independent claims 1 and 6.

Furthermore, Robinson in view of Leach does not teach or suggest the reserved address, "wherein the **instruction address is not known to the remote procedure call requestor**," as recited in independent claims 1 and 6. With regard to Robinson, an active task list table, which points to objects (remote procedure) is stored in allocated RAM of the requesting processor (call requestor) (Abstract and column 9, lines 22-38). Therefore, the instruction address of Robinson is known to the call requestor. With regard to Leach, the kernel, which is invoked by the client (call requester), determines the real method's address (instruction address) (column 11, lines 10-13). Therefore, the instruction address of Leach is known to the call requestor. In contrast to Robinson in view of Leach, the reserved address of the present invention enables a requestor to access and process a procedure remotely without having to know the instruction address. The requestor only needs to know and access the reserved address to initiate the software action. Since both Robinson and Leach teach that the instruction address is known to the call requestor, Robinson in view of Leach do not provide the benefits as with the present invention.

Accordingly, Robinson in view of Leach does not teach or suggest the *combination of steps* of the present invention, and these claims are allowable over Robinson in view of Leach.

Remaining dependent claims

Dependent claims 2-5 and 7-10 depend from claims 1 and 6, respectively. Accordingly, the above-articulated arguments related to claims 1 and 6 apply with equal force to claims 2-5 and 7-10, which are thus allowable over the cited references for at least the same reasons as claims 1 and 6.

Conclusion

In view of the foregoing, Applicants submit that claims 1-10 are patentable over the cited references. Applicants, therefore, respectfully request reconsideration and allowance of the claims as now presented.

Applicants' attorney believes that this application is in condition for allowance. Should any unresolved issues remain, the Examiner is invited to call Applicants' attorney at the telephone number indicated below.

Respectfully submitted,
SAWYER LAW GROUP LLP

March 19, 2004
Date

Joseph A. Sawyer, Jr.
Attorney for Applicant(s)
Reg. No. 30,801
(650) 493-4540